Tutorial On RISC V

Simulator

- I'll be using Rars, also see description of system calls here.
- Can try Venus, Github repo: https://github.com/kvakil/venus. Note that for system calls, their argument register is different, see this.

Examples

Hello Word

```
.globl start

.data

msg: .asciiz "hello world"

.text

start:
  li a0, 4
  la a1, msg
  ecall
  li a0, 10
  ecall
```

Fibonacci

```
.globl __start

.data
  msg1: .string "Please enter a number: "
  msg2: .string "The "
  msg3: .string " fibonnaci number is: "
```

```
.text

__start:
  li t0, 0
  li t1, 1
  # prints msg1
  li a0, 4
  la a1, msg1
  ecall
  # reads an int and moves it to register t3
  li a0, 5
  ecall
  mv t3, a0
  # prints a newline
  li a0, 11
  li a1, '\n'
  ecall
  # prints msg2
  li a0, 4
  la a1, msg2
  ecall
  # prints the int value in t3
  li a0, 1
  mv a1, t3
  ecall
  # fibonnaci program
fib:
  beq t3, zero, finish
  add t2, t1, t0
  mv t0, t1
  mv t1, t2
  addi t3, t3, -1
  j fib
finish:
  # prints msg3
  li a0, 4
  la a1, msg3
  ecall
  # prints the result in t0
  li a0, 1
  mv a1, t0
  ecall
  # prints a newline
  li a0, 11
  li a1, '\n'
```

```
ecall
# ends the program with status code 0
li a0, 10
ecall
```